

# **Zusammenfassung DASB**

## Data Science Basics

Maurin D. Thalmann

23. Januar 2020

# Inhaltsverzeichnis

<b>1 Learning Objectives</b>	<b>2</b>
<b>2 Introduction to Data Science</b>	<b>3</b>
2.1 Introductory Stizzle . . . . .	3
2.2 A Global View on Data Science . . . . .	3
2.3 Different Concepts in Data Analytics . . . . .	4
2.3.1 1993 - Online Analytical Processing (OLAP) . . . . .	4
2.3.2 1998 - Fuzzy Data Analysis . . . . .	4
2.3.3 2005 - Data Mining . . . . .	5
2.3.4 2013 - Predictive Modeling . . . . .	5
2.3.5 2013 - Data Driven Decision Making . . . . .	5
<b>3 Intro to R and Exploratory Data Analysis (EDA)</b>	<b>6</b>
3.1 Intro to R . . . . .	6
3.1.1 Basic Operation in R . . . . .	6
3.2 What is EDA . . . . .	6
3.2.1 EDA process „in practice“ . . . . .	7
3.3 Data Import / Transform / Link . . . . .	7
3.3.1 Data Import . . . . .	7
3.3.2 Data Transform (example dataset) . . . . .	7
3.3.3 Data Transform . . . . .	8
3.3.4 Data „Linkage“ . . . . .	8
3.3.5 Applying data loading/transforming/linking . . . . .	9
<b>4 ggplot2 and its usage in EDA</b>	<b>9</b>

# 1 Learning Objectives

## Introduction to Data Science

- Define the term „Data Science“ and differentiate it from similar concepts such as big data, data mining, OLAP, predictive modeling, data analysis
- Describe why and how value is created from data, especially in form of processes
- Describe the skills and activities of a data science professional
- Describe possible applications of data science

## Intro to R and Exploratory Data Analysis (EDA)

- Define the term „Exploratory Data Analysis“ and explain why it is a fundamental part of making data science
- Give a short introduction to R and present its main features
- Indicate the key actors and actions that compose EDA and how they are related to the data sources considered
- Indicate different ways of importing data into R for analysis and how to check for their quality
- Describe the main operations available in R (through libraries) to manipulate data and prepare it for the analysis

## ggplot2 and its usage in EDA

- Define the role of graphical representations in the „Exploratory Data Analysis“ and explain why it is a fundamental tool to make it effective
- Give a short introduction to ggplot2 and its features
- Ddescribe the main way of working of ggplot2 and how to compose complex graphs (e.g.: representing more than 3 variables in a single graph, for multidimensional data)
- Explain the role of the facets and how they can support the representation of „partial“ relationships
- Indicate the role of the coordinates in the graph usage for EDA

## 2 Introduction to Data Science

### 2.1 Introductory Stizzle

- More is different - with different quantity comes different quality, although „more“ is always dependent on the point of view.
- Data volume nowadays is exploding (276.12 billion GB of digital data)
- Data Science tries to close the gap between „Data available to an organization“ and „Percent of data an organization can process“.

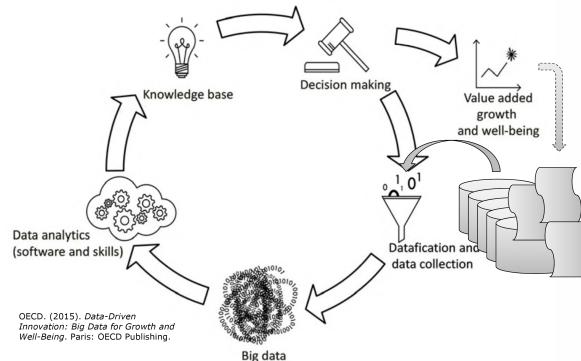


Abbildung 1: Beautiful illustration of the classic data value cycle

„**Data Science** is the extraction of actionable knowledge directly from data through a discovery, or hypothesis formulation and hypothesis testing“. Data Scientists generate knowledge from big data.

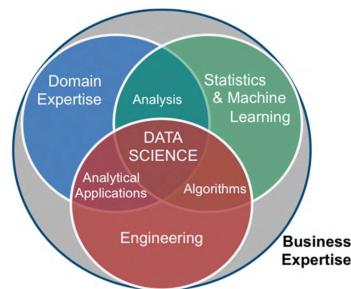


Abbildung 2: Skills needed in Data Science

### 2.2 A Global View on Data Science

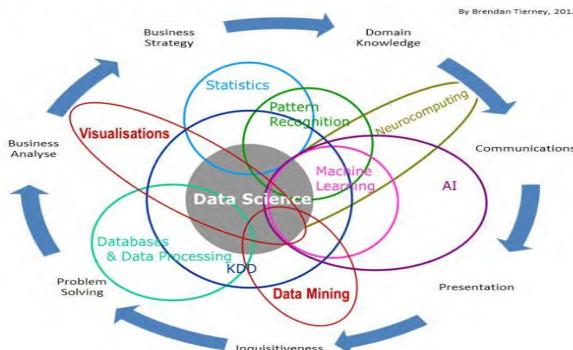


Abbildung 3: Data Science is multidisciplinary

- **AI**: programs that perform tasks resembling humans (learn and reason)
- **ML**: algorithms to learn from that data without explicit programming
- **DL**: subset of ML using artificial neural networks for treating vast amount of data (big data)
- **Data Science**: spans the collection, management, analysis and interpretation of large amounts of data with a wide range of applications → make informed decisions based on what was learned
- **EDA** (exploratory data analysis): extract insight from data (outside AI, human based)

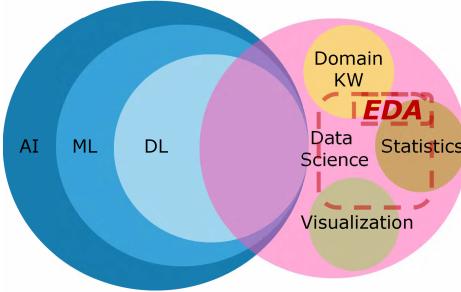
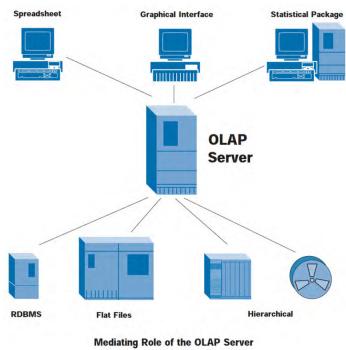


Abbildung 4: AI / ML / DL and Data Science

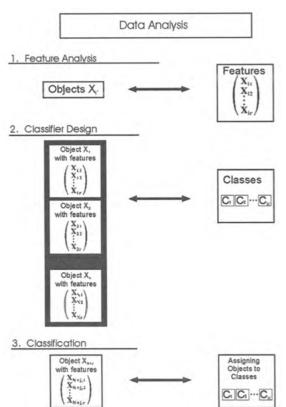
## 2.3 Different Concepts in Data Analytics

### 2.3.1 1993 - Online Analytical Processing (OLAP)



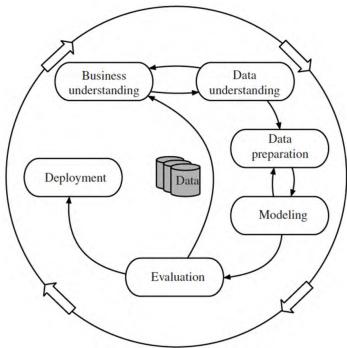
- It's online, not batch (interactive, not programmed in COBOL)
- The OLAP system should access the data required to perform the indicated analysis
- OLAP tools empower useranalysts to easily perform multi-dimensional analysis, which previously have been avoided because of their perceived complexity.

### 2.3.2 1998 - Fuzzy Data Analysis



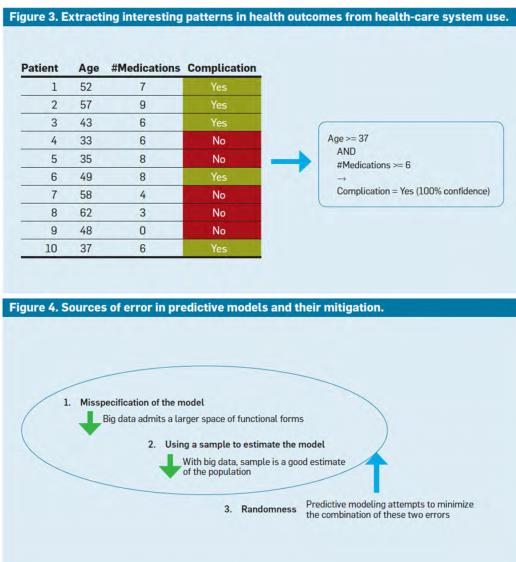
- Data analysis can be defined as **search for structure** in data
- In data analysis, objects are considered which are described by some attributes
- Most of the traditional methods for data analysis assume that patterns to be detected are two-valued
- Whenever this is not the case, the relationship between data and classes becomes gradual  
→ Fuzzy Classification

### 2.3.3 2005 - Data Mining



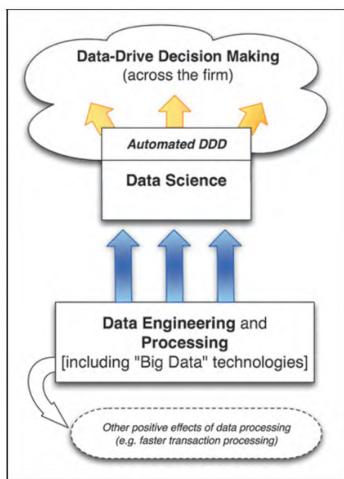
- Very similar concept to data science
- Machine Learning (modeling) is the technical core of practical data mining applications
- Data Mining is a business process related to *value* (finding the metaphorical gold nugget)
- The lifecycle of a data mining project is defined by the CRISP-DM reference model

### 2.3.4 2013 - Predictive Modeling



- A common epistemic requirement in assessing whether new knowledge is actionable for decision making in its *predictive power*, not just its ability to explain the past.
- The requirement on predictive accuracy on observations that will occur in the future is a key consideration in data science.

### 2.3.5 2013 - Data Driven Decision Making



- Data science involves much more than just data-mining algorithms.
- Successful data scientists must be able to view business problems from a *data perspective*.

### 3 Intro to R and Exploratory Data Analysis (EDA)

#### 3.1 Intro to R

- R ist a software and programming language for statistical applications and graphics
- It's an implementation of the programming language S, with some extensions
- It provides functions for linear/non-linear modelling, classical statistical tests, time-series analysis, classification, clustering etc.
- RStudio is an integrated development environment (IDE) for R

##### 3.1.1 Basic Operation in R

- Assignment:  
`my_var <- 46, nr <- 38`  
(You can also use `copy my_var = 46`, but it is not suggested)
- Printing (explicit and implicit): every line with only a name of an object  
Implicit: `my_var`  
Explicit using functions like `print(print(my_var))`
- Arithmetic operators:  
`4 + 5`  
`2 ^ 3 - 7 %% 3 - 2 ^ (3 - 7 %% 3)` etc. etc. etc.
- Control flow, conditionals, loops can be used as in any other language  
`for (i in 1:10) print(i)`  
`while (nr < 42) print(,We have "+str(nr)+,, : still space..."); nr=nr+1`

#### 3.2 What is EDA

- Exploratory Data Analysis is the
  - critical process of performing initial investigations on data so as to
    - \* discover patterns
    - \* spot anomalies
    - \* test hypothesis
    - \* check assumptions
  - with the help of **summary statistics** and **graphical representations**

The use of a statistical model is not mandatory, in this step, even if it is normally adopted to test hypotheses and check assumptions. For anomalies and patterns discovery, correlations (represented into the correlation matrix) and distribution (represented with box and whisker diagrams) are the usual approaches.

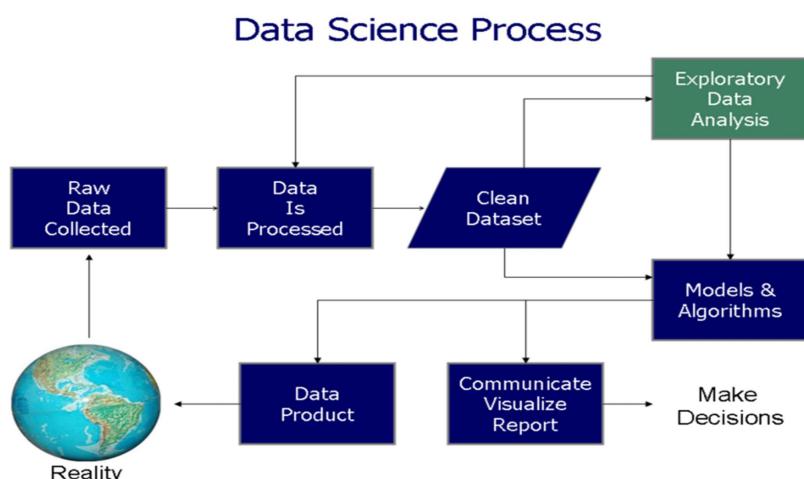


Abbildung 5: EDA in the context of Data Science

### 3.2.1 EDA process „in practice“

- EDA is an iterative cycle
- Process:
  - Generate questions about the data
  - Search for answers by visualizing, transforming and modelling the data
  - Use the insights to refine the questions and/or generate new questions
- Start with many ideas...
- some will be evidently useless and other will end up in a dead end
- a few of them will help to have a better understanding of the data/process
- repeat with generating other ideas until the insight is enough
- then you can apply the model to the data for forecasting and process explanation
- EDA also allows you to verify/test the quality of your data

## 3.3 Data Import / Transform / Link

### 3.3.1 Data Import

- R provides importing functions, depending on the format of the data source:
  - CSV / TSV / delimited texts: `read_csv()` & `read_csv2()` (semicolon), `read_tsv()`, `read_delim()`
  - Fixed width columns: `read_fwf()` & `read_table()`
  - Logs (e.g. webserver logs): `read_log()`

These are all similar in terms of parameters, examples:

```
1 read_delim(file , delim , quote = "\\", escape_backslash = FALSE, escape_double = TRUE,
2 2 col_names = TRUE, col_types = NULL, locale = default_locale() , na = c("", "NA") ,
3 3 quoted_na = TRUE, comment = "", trim_ws = FALSE, skip = 0, n_max = Inf, guess_max =
```

### 3.3.2 Data Transform (example dataset)

Example dataset loaded from: *Auto MPG Data Set*

```
1 > library(tidyverse)
2 > DS <- read.table("C:/Users/JumpStart/Downloads/auto-mpg.data",
3   quote="\\"", comment.char="")
4 > names(DS) <-
5   c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration",
6   "model year", "origin", "car name")
7 > DS <- as_tibble(DS); attach(DS); View(DS)
```

### 3.3.3 Data Transform

- Filter rows with `filter()`
  - subset observations based on their values  
`filter(DS, weight > 3500) or filter(DS, DS$weight > 3500)`
- Arrange rows with `arrange()`
  - changes the order of rows  
`arrange(DS, desc(DS$acceleration))`
- Select columns with `select()`
  - Subset columns with position and/or negative condition (by name or index)  
`select(DS, -1)`  
`select(DS, 2, 3, 4)`  
`select(DS, c(mpg, 5))`
- Add new variables with `mutate()`
  - Allow to create new dimension(s)  
`mutate(DS, a4w = acceleration/weight, pre75 = as.numeric(DS$model year' < 75))`
- Grouped summaries with `summarise()`
  - Allows to compute aggregate metrics  
`summarise(group_by(DS, 'model year'), mean_display = mean(displacement), sd_displ = sd(displacement), n=n())`
- All verbs work similarly:
  - The first argument is a dataframe
  - The subsequent arguments describe what to do with the data frame, using the variable names (without quotes)
  - The result is a new data frame

### 3.3.4 Data „Linkage“

Some prices of cars of the 70's: *70's Cars*.

This data is imported into a new tibble and we will try to use it for creating another View for exploring the relationship between cost, year, power and mpg.

```

1 > car_cost <- read_delim("C:/Users/JumpStart/switchdrive/__/DASB/SW02/car_197x_costs_2.csv",
2   delim = "_",
3   escape_double = FALSE,
4   na = "NA",
5   trim_ws = TRUE,
6   col_types = cols('Car Name' = col_factor(),
7     Price = col_character(),
8     Location = col_character(),
9     'Matriculation Year' = col_integer()))
10 > car_cost <- as_tibble(car_cost)
11 > car_cost <- mutate(car_cost, Location = sapply(Location, as.factor))
12 > car_cost <- mutate(car_cost, Price = as.integer(str_replace(str_sub(Price, 2, -1), ",","")))
13 > View(car_cost)

```

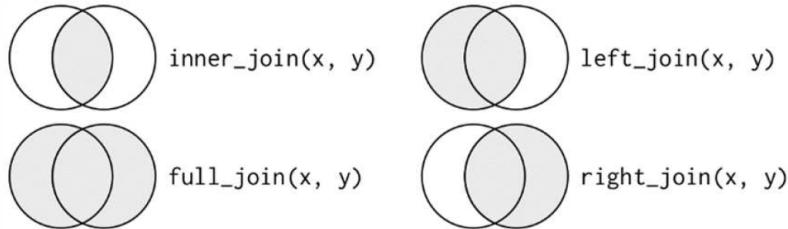


Abbildung 6: R functions to link data frames with each other

There are different types of R functions for linking tibbles/dataframes (~datasets)

- **Mutating** joins allows to combine variable from two tables
  - `inner_join(x, y, by = keys)`  
matches pairs of observations whenever their keys are equal
  - Outer join keeps observations that appear in at least one of the tables. There are three types of outer joins:
    - \* `left_join(x, y, by = keys)`  
keeps all observations in x
    - \* `right_join(x, y, by = keys)`  
keeps all observations in y
    - \* `full_join(x, y, by = keys)`  
keeps all observations in x and y
- **Filtering** joins allow to restrict the set of row from the 1<sup>st</sup> table
  - `semi_join(x, y, by = keys)`  
keeps all observations in x that have a match in y
  - `anti_join(x, y, by = keys)`  
drops all observations in x that have a match in y
- **Set Operations** (work on complete rows, comparing each variable values. Inputs needs the same variables, for treating observations like sets)
  - `intersect(x, y)`  
return only observations in both x and y
  - `union(x, y)`  
return unique observations in x and y
  - `setdiff(x, y)`  
return observations in x, but not in y

### 3.3.5 Applying data loading/transforming/linking

- Find one or more data sources you are interested into (UCI ML Repository / Kaggle)
- Start the EDA „mindset“:
  1. What I believe about the data / what I would like to discover  
→ Make hypotheses and formulate assumptions
  2. Think how this can be supported by data → experiment
  3. Run the experiment to verify
  4. Acquire new knowledge that can help you improve step 1
- Repeat the process at least 3 times  
*(Optional: short report of the process you followed, document it in a short presentation)*

## 4 ggplot2 and its usage in EDA